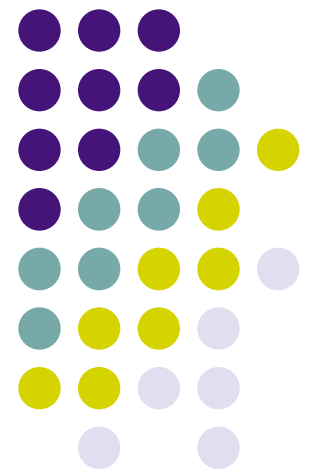
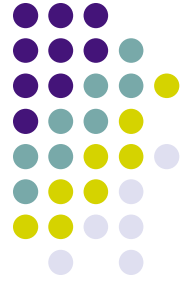


The NASA Workflow Tool: Project Overview

April 23, 2009





What is a Workflow Tool?

A system that provides end-to-end support for setting up and running complex model experiments.

Enable users to concentrate on the scientific results rather than the system environments.



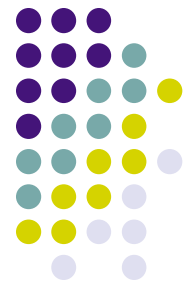
Overview of the NASA Workflow Tool



- Modern GUI tool that provides clear display of options
- Easily configure, run and monitor model experiments
- Detect configuration errors before submitting them
- Automate the complex tasks required to run
- Save experiment configurations for repeatable experiments and sharing experiments between users

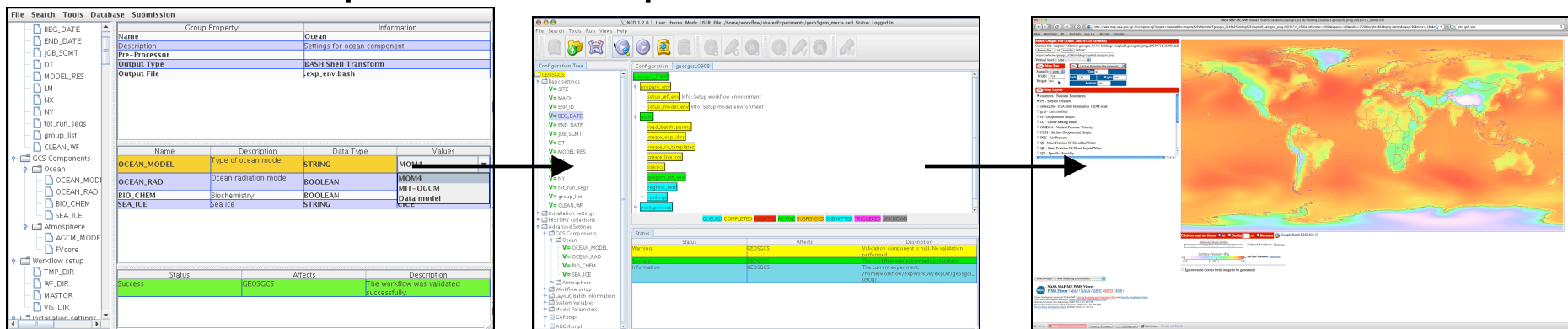
Name	Description	Data Type	Values
OCEAN_MODEL	Type of ocean model	STRING	MOM4
OCEAN_RAD	Ocean radiation model	BOOLEAN	MOM4
BIO_CHEM	Biochemistry	BOOLEAN	MIT- OGCM
SEA_ICE	Sea ice	STRING	Data model
			CICE

Goal: *Simplify tedious work needed for scientists to perform complex model runs and data management*

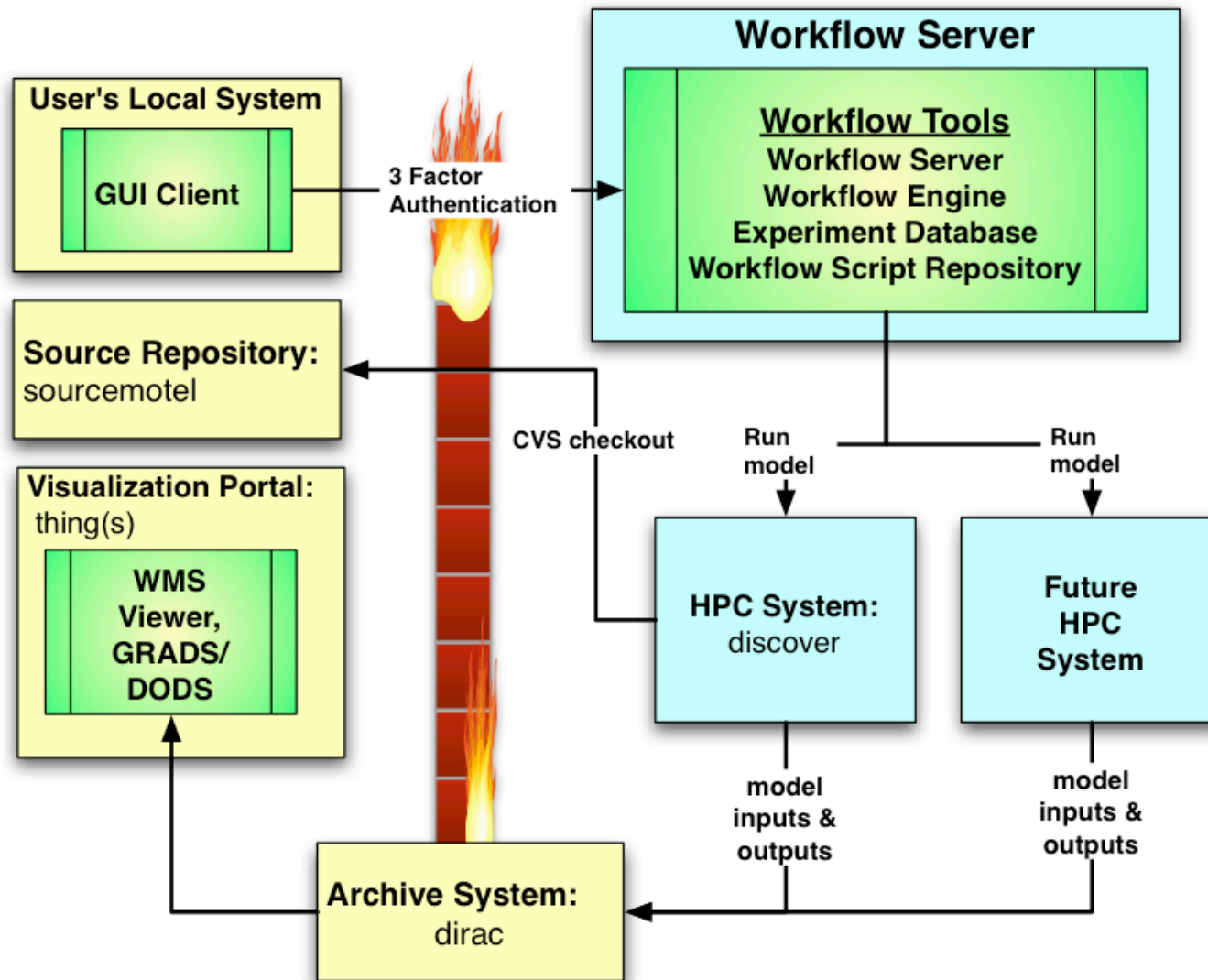


Benefits to End-Users

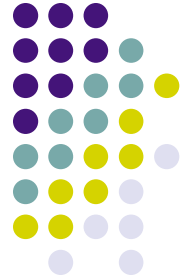
- Remove user from having to manage the system environment
- Clearly present model and runtime options
- Detect common configuration errors early
- Automate the complex tasks underneath
- Provide other tools through an end-to-end chain (configuration to monitor/control to visualization/analysis)
- Make experiments repeatable and shareable



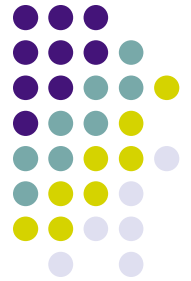
Workflow System at NASA



- Run many types of workflows from a single location
- Connects to HPC resources under the hood in order to perform tasks
- Works under high-security environments



NASA Experiment Designer & Light-Weight Workflow Engine



NED & LWWE

- Client/server
- Componentized, pluggable architecture for adding new features
- Written in Java (portable)
- XML data formats
- Security features
 - Uses encrypted SSL socket communication
 - Performs LDAP and Radius authentication with NCCS servers

Workflow Tool Components



File Search Tools Database Submission

Group Property		Information	
Name		Ocean	
Description		Settings for ocean component	
Pre-Processor			
Output Type		BASH Shell Transform	
Output File		exp_env.bash	

Name	Description	Data Type	Values
OCEAN_MODEL	Type of ocean model	STRING	MOM4
OCEAN_RAD	Ocean radiation model	BOOLEAN	MOM4
BIO_CHEM	Biochemistry	BOOLEAN	MIT-OGCM
SEA_ICE	Sea ice	STRING	Data model

Status	Affects	Description
Success	GEOSGCS	The workflow was validated

NED Client

NED 1.2.0.3 User: rburns Mode: USER File: /home/workflow/sharedExperiments/geos5gcm_merra.ned Status: Logged In

File Search Tools Run Views Help

Configuration Tree

- GEOSGCS
 - Basic settings
 - V= SITE
 - V= MACH
 - V= EXP_ID
 - V= BEG_DATE
 - V= END_DATE
 - V= JOB_SGMT
 - V= DT
 - V= MODEL_RES
 - V= LM
 - V= NX
 - V= NY
 - V= tot_run_segs
 - V= group_list
 - V= CLEAN_WF
 - HISTORY collections
 - Advanced Settings
 - GCS Components
 - Ocean
 - OCEAN_MODEL
 - OCEAN_RAD
 - BIO_CHEM
 - SEA_ICE
 - Atmosphere
 - Workflow setup
 - TMP_DIR
 - WF_DIR
 - MASTOR
 - VIS_DIR
 - Installation settings
 - System variables
 - Model Parameters
 - CAP.tmpl
 - AGCM.tmpl

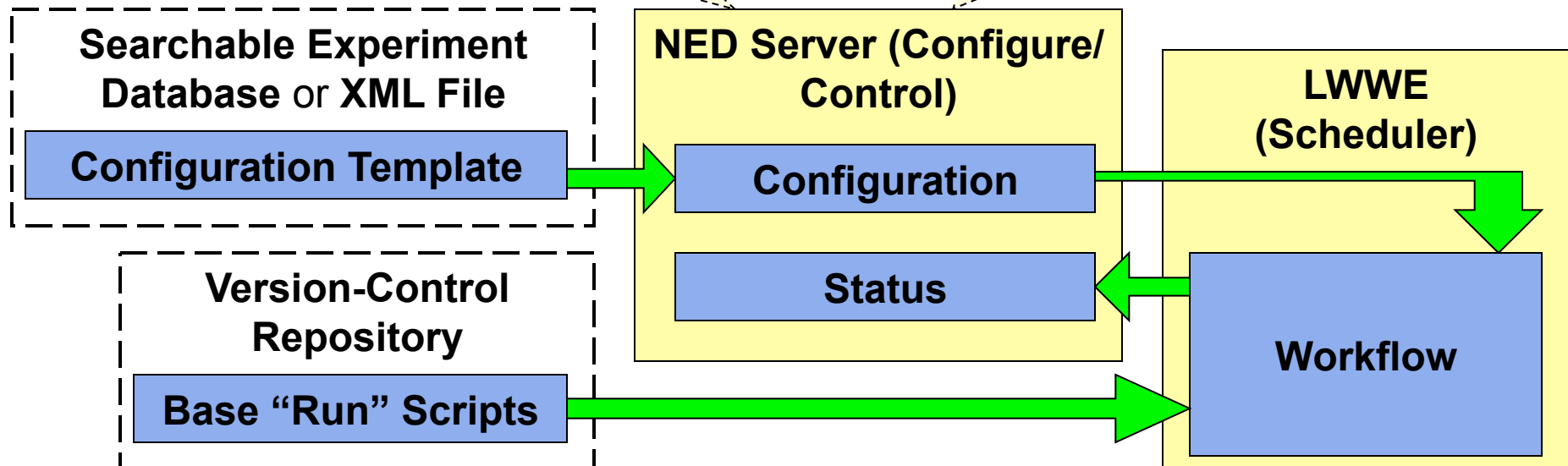
Configuration: geosgcs_0908

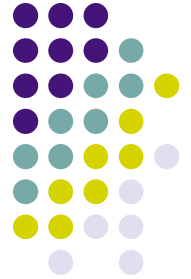
- prepare_env
- setup_wf_env
- setup_model_env
- load_batch_params
- create_exp_dir
- create_rc_templates
- create_live_rc
- linkbc
- progress_bar
- runloop
- post_process

STATUS: QUEUED COMPLETED FAILED SUSPENDED SUBMITTED TRIGGERED UNKNOWN

Status	Affects	Description
Warning	GEOSGCS	Validation component is null. No validation performed
Information	GEOSGCS	The current experiment: /home/workflow/expWorkDir/expDir/geosgcs_0908/

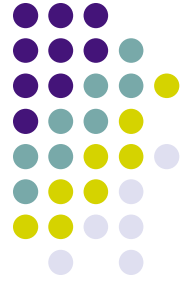
NED Client





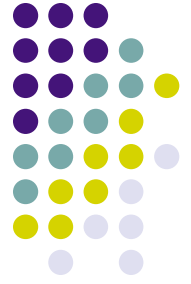
GEOS5 Workflow:

Recent Work



GEOS-5 CVS Tags

- Added support for:
 - GEOSagcm-Eros_7_24
 - GEOSagcm-Eros_7_25
 - Other tags can be tested, but not yet fully supported
- Tagged experiments can be checked out, built and run from the Workflow tool
- Obtained restart files for supported tags
- Workflow automatically creates the build and run environments according to tag



GEOS5 Fortuna Workflow

- Now support Fortuna-1_3 tag from the Fortuna module of GEOS5
- User selects the CVS module (either GEOSagcm or Fortuna)
- Fortuna compatible restart files acquired and archived
- New aeroclim, optics and other boundary conditions supported
- Added support for .nc4 file formats

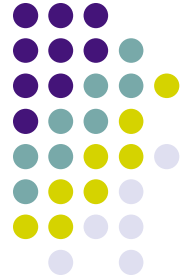
GEOS5 Workflow



- “Pythonization” of GEOS5 workflow scripts
 - Converted from bash scripts currently used in workflow
 - Why? Increased flexibility, reuse, error handling and execution security
 - Updated scripts per code review from the team
 - Tested workflow for the comparison with the original workflow used in bash
- See the next slide for the example

GEOS5 Workflow

Showing python and bash scripts for the comparison



```
#-----  
# Performs the notification task that has option of emailing.  
# Exception will be thrown if any failure of email or missing key.  
#-----
```

```
def notification(self):  
    # Check for the email option  
    print "\nEMAIL_OPTION: %s" % self.vars['EMAIL_OPTION']  
    if self.vars['EMAIL_OPTION'] != 'true':  
        print "INFO: no emailing"  
        return  
  
    # Set up the email  
    print "\nStarting to email for notification..."  
    nedID = self.vars['NED_UNIQUE_ID']  
    email = "%s@nccs.nasa.gov" % self.vars['NED_REAL_USER']  
    echoCmd = "echo -e \"Subject: %s\\n%s is completed.\\n\" % (nedID,nedID)"  
    mailCmd = "/usr/bin/mail " + email  
    sysCmd = "%s | %s" % (echoCmd, mailCmd)  
    errCode = os.system(sysCmd)  
    print "\nStatus for %s: %d" % (sysCmd, errCode)  
    if errCode:  
        raise EnvironException("Failure of notification")  
    print "\nDone with notification."
```

```
...  
....  
. $NED_WORKING_DIR/.exp_env.bash
```

```
# Check for the email option  
echo -e "\nEMAIL_OPTION: $EMAIL_OPTION"  
if [ "$EMAIL_OPTION" != "true" ]; then  
    echo "INFO: no emailing"  
    exit 0  
fi
```

```
# Create the script file for email  
echo -e "\nCreating the script file for emailing"  
EMAIL_FILE=email.bash  
EMAIL_PATH=$NED_WORKING_DIR/$NED_UNIQUE_ID/bin  
EMAIL_FULL=$EMAIL_PATH/$EMAIL_FILE
```

```
echo "EMAIL_FILE: $EMAIL_FILE"  
echo "EMAIL_PATH: $EMAIL_PATH"  
echo "EMAIL_FULL: $EMAIL_FULL"
```

```
echo "#!/bin/bash" > $EMAIL_FULL  
echo "export NED_REAL_USER=$NED_REAL_USER" >> $EMAIL_FULL  
echo "export NED_UNIQUE_ID=$NED_UNIQUE_ID" >> $EMAIL_FULL  
echo "export SUBJECT=$NED_UNIQUE_ID" >> $EMAIL_FULL  
echo "export EMAIL=$NED_REAL_USER@nccs.nasa.gov" >> $EMAIL_FULL  
echo "echo \"Workflow '$NED_UNIQUE_ID' is complete.\" | /usr/bin/mail -s \"\\n\\n$SUBJECT\" \"$EMAIL\" >> $EMAIL_FULL"  
chmod +x $EMAIL_FULL  
echo -e "Done with creating the script file for emailing"  
  
....  
....
```

GEOS 5 Workflow



- Current development – Lightweight Workflow Engine (LWWE) conversion
 - Created the workflow architecture in XML and tested simple tasks including scripts.
 - Continue with converting the GEOS5 Workflow used in LWWE from SMS.
 - Why to use LWWE?
 - Flexibility and improved performance